WHAT IS CLAIMED IS:

1. A computer-based method, comprising:

monitoring substantially in parallel a plurality of subsystems of the operating system during runtime for an event caused from a request made by a Downloadable;

interrupting processing of the request;

comparing information pertaining to the Downloadable against a predetermined security policy; and

performing a predetermined responsive action based on the comparison.

2. The method of claim 1, wherein monitoring the operating system includes monitoring a request sent to a Downloadable engine.

3. The method of claim 2,

wherein the Downloadable engine includes a Java^{TM} virtual machine having Java ^{TM} classes; and

wherein monitoring the operating system includes monitoring each Java^{TM} class for receipt of the request.

4. The method of claim 2,

wherein the Downloadable engine includes an AppletX^{TM} platform having a message engine, a dynamic-data-exchange and a dynamically-linked library; and

wherein monitoring the operating system includes monitoring the message engine, the dynamic-data-exchange and the dynamically-linked library for receipt of the request.

5. The method of claim 1, further comprising determining whether information pertaining to the Downloadable violates a security rule.

6. The method of claim 5, further comprising determining whether violation of the security rule violates the security policy.

7. The method of claim 1, further comprising:

comparing information pertaining to the Downloadable with information pertaining to a predetermined suspicious Downloadable; and

performing a predetermined responsive action based on the comparison with the information pertaining to the predetermined suspicious Downloadable.

8. The method of claim 1, wherein the predetermined responsive action includes storing results of the comparison in an event log.

9. The method of claim 1, wherein the predetermined responsive action includes informing the user when the security policy has been violated.

10. The method of claim 1, wherein the predetermined responsive action includes storing information on the Downloadable in a suspicious Downloadable database.

11. The method of claim 1, wherein the predetermined responsive action includes discarding the Downloadable.

12. A system, comprising:

a security policy;

a plurality of operating system interfaces operating substantially in parallel, each interface for recognizing a runtime event in a subsystem of the operating system caused from a request made by a Downloadable;

a first comparator coupled to the interfaces for comparing information pertaining to the received Downloadable with the security policy; and a response engine coupled to the first comparator for performing a predetermined responsive action based on the comparison with the security policy.

13. The system of claim 12, wherein the interfaces include a Java™ class extension for monitoring a Java™ class in a Java™ virtual machine for receipt of a request.

14. The system of claim 12, wherein the interfaces include an AppletX™ extension for monitoring a message engine, a dynamic-data-exchange and a dynamically-linked library in an AppletX™ environment for receipt of a request.

15. The system of claim 12, further comprising

a security rule; and

a second comparator, coupled to the interfaces and to the response engine, for determining whether information pertaining to the Downloadable violates the security rule.

16. The system of claim 15, wherein the first comparator determines whether violation of the security rule violates the security policy.

17. The system of claim 12, further comprising

a predetermined suspicious Downloadable; and

a second comparator coupled to the interfaces for comparing information pertaining to the Downloadable with information pertaining to the predetermined suspicious Downloadable;

wherein the response engine is further coupled to the second comparator and performs the responsive action based on the comparison with the information pertaining to the predetermined suspicious Downloadable.

18. The system of claim 12, further comprising an event log coupled to the first comparator for storing results of the comparison.

19. The system of claim 12, further comprising a user interface coupled to the first comparator.

20. The system of claim 12, further comprising a suspicious Downloadable database for storing information on known and previously-deemed suspicious Downloadables.

21. The system of claim 12, wherein the predetermined suspicious action includes discarding the Downloadable.

22. A system for determining whether a Downloadable, which is received by a Downloadable engine, is suspicious, comprising:

means for monitoring substantially in parallel a plurality of subsystems of the operating system during runtime for an event caused from a request made by a Downloadable;

means for interrupting processing of the request;

means for comparing information pertaining to the Downloadable against a predetermined security policy; and

means for performing a predetermined responsive action based on the comparison.

23. The system of claim 22, wherein the means for monitoring the operating system includes means for monitoring a request sent to a Downloadable engine.

24. The system of claim 23,

wherein the Downloadable engine includes a Java™ virtual machine having Java™ classes; and wherein the means for monitoring the operating system includes means for monitoring each Java^{TM} class for receipt of the request.

25. The system of claim 23,

wherein the Downloadable engine includes an AppletX^{TM} platform having a message engine, a dynamic-data-exchange and a dynamically-linked library; and

wherein the means for monitoring the operating system includes means for monitoring the message engine, the dynamic-data-exchange and the dynamically-linked library for receipt of the request.

26. The system of claim 22, further comprising means for determining whether information pertaining to the Downloadable violates a security rule.

27. The system of claim 26, further comprising means for determining whether violation of the security rule violates the security policy.

28. The method of claim 22, further comprising:

means for comparing information pertaining to the Downloadable with information pertaining to a predetermined suspicious Downloadable; and

means for performing a predetermined responsive action based on the comparison with the information pertaining to the predetermined suspicious Downloadable.

29. The system of claim 22, wherein the predetermined responsive action includes storing results of the comparison in an event log.

30. The system of claim 22, wherein the predetermined responsive action includes informing the user when the security policy has been violated.

31. The system of claim 22, wherein the predetermined responsive action includes storing information on the Downloadable in a suspicious Downloadable database.

32. The system of claim 22, wherein the predetermined responsive action includes discarding the Downloadable.

33. A computer-readable storage medium storing program code for causing a computer to perform the steps of:

monitoring substantially in parallel a plurality of subsystems of the operating system during runtime for an event caused from a request made by a Downloadable;

interrupting processing of the request;

comparing information pertaining to the Downloadable against a predetermined security policy; and

performing a predetermined responsive action based on the comparison.

34. The medium of claim 33, wherein monitoring the operating system includes monitoring a request sent to a Downloadable engine.

35. The medium of claim 33,

wherein the Downloadable engine includes a Java$^{\text{TM}}$ virtual machine having Java$^{\text{TM}}$ classes; and

wherein monitoring the operating system includes monitoring each Java$^{\text{TM}}$ class for receipt of the request.

36. The medium of claim 35,

wherein the Downloadable engine includes an AppletX$^{\text{TM}}$ platform having a message engine, a dynamic-data-exchange and a dynamically-linked library; and wherein monitoring the operating system includes monitoring the message engine, the dynamic-data-exchange and the dynamically-linked library for receipt of the request.

37. The medium of claim 33, further comprising determining whether information pertaining to the Downloadable violates a security rule.

38. The medium of claim 37, further comprising determining whether violation of the security rule violates the security policy.

39. The medium of claim 33, further comprising:

comparing information pertaining to the Downloadable with information pertaining to a predetermined suspicious Downloadable; and

performing a predetermined responsive action based on the comparison with the information pertaining to the predetermined suspicious Downloadable.

40. The medium of claim 33, wherein the predetermined responsive action includes storing results of the comparison in an event log.

41. The medium of claim 33, wherein the predetermined responsive action includes informing the user when the security policy has been violated.

42. The medium of claim 33, wherein the predetermined responsive action includes storing information on the Downloadable in a suspicious Downloadable database.

43. The medium of claim 33, wherein the predetermined responsive action includes discarding the Downloadable.

44. The system of claim 1, wherein each subsystem includes one of a file system, network system, process system or memory system.

45. The system of claim 12, wherein each subsystem includes one of a file system, network system, process system or memory system.

46. The system of claim 22, wherein each subsystem includes one of a file system, network system, process system or memory system.

47. The system of claim 33, wherein each subsystem includes one of a file system, network system, process system or memory system.

add
a